

LISTING OF THE CLAIMS

1. (Previously Presented) A computer implemented method for generating embedded code from a graphical model, the method comprising:

prompting a user to specify at least one code generation goal from a plurality of code generation goals, the at least one code generation goal being used to generate embedded code from the graphical model in a graphical modeling environment, the at least one code generation goal relating to a characteristic of the code to be generated from the graphical model, the graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations, the graphical model being capable of simulation based on the equations;

changing parameters of the graphical model that are inconsistent with the at least one code generation goal; and

generating embedded code in accordance with the at least one code generation goal.

2. (Canceled)

3. (Previously Presented) The method of claim 1, further comprising:

providing feedback to the user regarding compliance of the graphical model with a selected condition.

4. (Original) The method of claim 3, wherein the user selects the selected condition through a user interface.

5. (Original) The method of claim 4, wherein the user interface displays a list of conditions to be checked, and prompts the user to select one or more of the conditions.

6. (Previously Presented) The method of claim 3, wherein providing feedback to the user regarding the compliance of the graphical model with the selected condition further comprises:

displaying a hyperlink that links the selected condition to an object of the graphical model that does not comply with the selected condition.

7. (Previously Presented) The method of claim 3, further comprising:
modifying an object of the graphical model that does not comply with the selected condition.
8. (Previously Presented) The method of claim 7, wherein modifying further comprises:
identifying the object; and
prompting the user to manually modify a parameter of the object.
9. (Previously Presented) The method of claim 7, wherein modifying further comprises:
automatically modifying a parameter of the graphical model to comply with the selected condition.
10. (Original) The method of claim 1, wherein the graphical model is a block diagram.
11. (Original) The method of claim 1, wherein each code generation goal corresponds to a general code generation goal.
12. (Previously Presented) The method of claim 11, further comprising:
prompting the user to specify at least one detailed code generation goal for each specified general code generation goal.
13. (Previously Presented) The method of claim 12, further comprising:
configuring the graphical model to comply with each detailed code generation goal.
14. (Previously Presented) A computer-implemented method of preparing a graphical model for embedded code generation, the method comprising:

acquiring a code generation goal, the code generation goal being used to generate embedded code from the graphical model, the code generation goal relating to a characteristic of the code to be generated from the graphical model, the graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations, the graphical model being capable of simulation based on the equations;

changing parameters of the graphical model that are inconsistent with the code generation goal; and

generating embedded code in accordance with the code generation goal.

15. (Previously Presented) The method of claim 14, further comprising:
identifying a condition that does not comply with the code generation goals.

16. (Previously Presented) A computer-implemented method of preparing a graphical model for embedded code generation, the method comprising:

acquiring at least one code generation goal, the acquired at least one code generation goal being used to generate embedded code from the graphical model, the acquired at least one code generation goal relating to a characteristic of the code to be generated from the graphical model, the graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations, the graphical model being capable of simulation based on the equations;

displaying information regarding compliance of the graphical model with the acquired at least one code generation goal;

changing parameters of the graphical model that are inconsistent with the acquired at least one code generation goal; and

generating embedded code in accordance with the acquired at least one code generation goal.

17. (Canceled)

18. (Previously Presented) The method of claim 16, wherein displaying information further comprises:

acquiring at least one condition; and

displaying information regarding compliance of the graphical model with the acquired at least one condition.

19. (Previously Presented) The method of claim 18, wherein the condition is acquired from a user through a user interface.

20. (Original) The method of claim 19, wherein the user interface displays a list of conditions to be checked, and prompts the user to select one or more of the conditions.

21. (Previously Presented) The method of claim 18, wherein displaying information regarding compliance of the graphical model with the acquired at least one condition comprises:

displaying a hyperlink that links the selected condition to an object of the graphical model that does not comply with the acquired condition.

22. (Previously Presented) The method of claim 18, further comprising:

modifying an object of the graphical model that does not comply with the acquired condition.

23. (Previously Presented) The method of claim 22, wherein modifying comprises:

identifying the object and prompting the user to manually modify a parameter of the object.

24. (Previously Presented) The method of claim 22, wherein modifying comprises:

automatically modifying a parameter of the graphical model to comply with the acquired condition.

25. (Original) The method of claim 16, wherein the graphical model is a block diagram.

26. (Canceled)

27. (Previously Presented) The method of claim 16, wherein each code generation goal corresponds to a general code generation goal.

28. (Previously Presented) The method of claim 27, further comprising:
prompting the user to specify at least one detailed code generation goal for each specified general code generation goal.

29. (Previously Presented) The method of claim 28, further comprising:
configuring the graphical model to comply with each detailed code generation goal.

30. (Previously Presented) In a graphical modeling environment, a computer-readable medium holding computer-executable instructions the medium holding:

one or more instructions for acquiring at least one code generation goal, the acquired at least one code generation goal being used to generate embedded code from the graphical model, the acquired at least one code generation goal relating to a characteristic of the code to be generated from the graphical model, the graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations, the graphical model being capable of simulation based on the equations;

one or more instructions for displaying information regarding compliance of the graphical model with the acquired at least one code generation goal;

one or more instructions for changing parameters of the graphical model that are inconsistent with the acquired at least one code generation goal; and

one or more instructions for generating embedded code in accordance with the acquired at least one code generation goal.

31. (Previously Presented) In a graphical modeling environment, a computer-readable medium holding computer-executable instructions the medium holding:

one or more instructions for acquiring a code generation goal, the code generation goal being used to generate embedded code from the graphical model, the code generation goal relating to a characteristic of the code to be generated from the graphical model, the graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations, the graphical model being capable of simulation based on the equations;

one or more instructions for changing parameters of the graphical model that are inconsistent with the code generation goal; and

one or more instructions for generating embedded code in accordance with the code generation goal.

32. (Previously Presented) In a graphical modeling environment, a computer-readable medium holding computer-executable instructions the medium holding:

one or more instructions for prompting a user to specify at least one code generation goal from a plurality of code generation goals, the at least one code generation goal being used to generate embedded code from the graphical model in a graphical modeling environment, the at least one code generation goal relating to a characteristic of the code to be generated from the graphical model, the graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations, the graphical model being capable of simulation based on the equations;

one or more instructions for changing parameters of the graphical model that are inconsistent with the at least one code generation goal; and

one or more instructions for generating embedded code in accordance with the at least one code generation goal.

33. (Previously Presented) An apparatus comprising:
at least one processor;

a memory coupled to the at least one processor; and

a computer program residing in the memory and being executed by the at least one processor, wherein the computer program includes a wizard for guiding a user through a process for preparing a graphical model for a code generation process for creating code based on the graphical model and at least one code generation goal, wherein the at least one code generation goal relates to a characteristic of the code, the graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations, the graphical model being capable of simulation based on the equations; wherein the wizard configures the graphical model based on the at least one code generation goal, wherein the computer program generates code in compliance with the at least one code generation goal.

34. (Canceled)

35. (Canceled)

36. (Canceled)

37. (Previously Presented) The apparatus of claim 33, wherein the wizard prompts the user to select one or more conditions to be checked in the graphical model.

38. (Previously Presented) The apparatus of claim 37, wherein the wizard identifies objects in the graphical model that do not comply with the selected conditions.

39. (Previously Presented) The apparatus of claim 37, wherein the wizard modifies objects in the graphical model that do not comply with the selected conditions.

40. (Previously Presented) The method of claim 1, wherein the at least one code generation goal is selected from the group consisting of:

a target application code generation goal;

- a maximum efficiency code generation goal;
- an aspect of memory code generation goal;
- an auto-generated identifier code generation goal;
- a comment code generation goal;
- an interface code generation goal;
- a model combination code generation goal; and
- a HyperText Markup Language report code generation goal.

41. (Previously Presented) The method of claim 1, wherein the at least one code generation goal is selected from the group consisting of:

- a set of predefined options for a target application code generation goal comprising a floating point target application option, a mixed point target application option, and a fixed point target application option.

- a set of predefined options for a maximum efficiency code generation goal comprising a maximum efficiency option and a non-maximum efficiency option;

- a set of predefined options for an aspect of memory code generation goal comprising a Random Access Memory (RAM) option and a Read-Only Memory (ROM) option;

- a set of predefined options for an auto-generated identifier code generation goal comprising a verbose auto-generated identifier option and a non-verbose auto-generated identifier option;

- a set of predefined options for a traceability code generation goal comprising a comments in the code option and a no comments in the code option;

- a set of predefined options for an interface code generation goal comprising a passing data as arguments option and a not passing data as arguments option;

- a set of predefined options for a model combination code generation goal comprising a single model option and a multiple model option; and

- a set of predefined options for a reporting code generation goal comprising an option for including a HTML report with the generated code and an option for not including a HTML report with the generated code.